**DOCUMENT RESUME**

ED 062 888 FL 003 229

| | |
|---|---|
| AUTHOR | Lehmann, Winfred P.; Stachowitz, Rolf A. |
| TITLE | Development of German-English Machine Translation System. |
| INSTITUTION | Texas Univ., Austin. Linguistics Research Center. |
| SPONS AGENCY | Rome Air Development Center, Griffiss AFB, N.Y. |
| REPORT NO | RADC-TR-72-47 |
| PUB DATE | Mar 72 |
| NOTE | 58p. |
| EDRS PRICE | MF-$0.65 HC-$3.29 |
| DESCRIPTORS | *Computational Linguistics; *English; *German; Lexicography; *Linguistic Theory; *Machine Translation; Semantics; Syntax; Systems Development; Translation |

ABSTRACT

The report presents a progress in theoretical linguistics, descriptive linguistics, lexicography, and systems design in the development of a German-English Machine Translation System. Work in the theoretical group concentrated on intrasentenial disambiguation and on improving certain parts of the system to achieve greater economy in processing. The linguistic group was engaged in correcting and updating the existing German and English lexical data bases by assigning syntactic and semantic selection restrictions to lexical items. Work in the system group concentrated on the reduction of the size of the existing linguistics research system lexical data base without information loss, on the conversion of this data base to the linguistics research system script format, on the construction of supporting programs to expedite and facilitate the updating of the linguistics research system word lists, and on the construction of part of the linguistics research system grammar maintenance and systems programs. (Author)

ED 062 888

RADC-TR-72-47
Technical Report
March 1972

# DEVELOPMENT OF GERMAN-ENGLISH MACHINE TRANSLATION SYSTEM

University of Texas at Austin

Approved for public release;
distribution unlimited.

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
OFFICE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL OFFICE OF EDUCATION POSITION OR POLICY.

Rome Air Development Center
Air Force Systems Command
Griffiss Air Force Base, New York

FL 003 225

# DEVELOPMENT OF GERMAN-ENGLISH MACHINE TRANSLATION SYSTEM

Dr. Winfred P. Lehmann
Dr. Rolf A. Stachowitz

University of Texas at Austin

Approved for public release;
distribution unlimited.

## FOREWORD

This technical report is the First Annual Report by the University of Texas at Austin, Linguistics Research Center, Austin, Texas, under contract F30602-70-C-0118, Job Order Number 45940000, for Rome Air Development Center, Griffiss Air Force Base, New York. It covers the period from 1 February 1970 to 31 January 1971. Sgt. Charles S. Bond, Jr. (IRDT) is the RADC Project Engineer.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS).

This technical report has been reviewed and is approved.

Approved: Charles S. Bond Jr.
CHARLES S. BOND, JR., Sgt,USAF
Technical Evaluator

Approved: Maj John Mantak
WILLIAM B. MOORE
Chief, Intel Data Handling Branch
Intel & Recon Division

*ABSTRACT*

Research in theoretical linguistics, descriptive linguistics, lexicography, and systems design pertinent to the Linguistics Research System for mechanical translation performed at the Linguistics Research Center is described. Work in the theoretical group concentrated on intra-sentential disambiguation and on improving certain parts of the system to achieve greater economy in processing. The linguistic group was engaged in correcting and updating the existing German and English lexical data bases by assigning syntactic and semantic selection restrictions to lexical items. Work in the systems group concentrated on the reduction of the size of the existing LRS lexical data base without information loss, on the conversion of this data base to the LRS subscript format, on the construction of supporting programs to expedite and facilitate the updating of the LRS word lists, and on the construction of part of the LRS grammar maintenance and systems programs.

# TABLE OF CONTENTS

*INTRODUCTION*

I. *THEORY*: The Linguistics Research System

1.1 Canonical Forms I-1
1.2 Normal Forms I-2
1.3 Mechanical Translation I-2
1.4 Subscript Grammars I-3
1.5 Syntactic Grammars I-4
1.6 Normal Form Grammars I-4
1.7 Analysis Procedure I-5
1.8 Intra-Sentential Disambiguation I-6
1.9 Changes in Subscript Grammar Format and Storage I-6
1.10 Example for Intra-Sentential Disambiguation I-8

II. *LEXICOGRAPHY*

2.1 Existing Data II-1
2.2 Progress II-4
2.3 Development of a General Classification System II-15

III. *PROGRAMMING*

3.1 Grammar Conversion Programs III-1
3.2 Systems Programs III-5
3.3 Supporting Programs III-5
3.4 Program Descriptions III-6

*CONCLUSION*

*References*

# INTRODUCTION

The difficulties that confront attempts to mechanically recognize and produce sentences in natural language generally arise from two causes. One is the lack of a lexicon with precise information on the syntactic and semantic properties of the context in which these lexical items may occur. The other source of difficulty is the concomitant generality of the recognition grammars which is necessary in order to keep the number of required rules to a manageable size. As a result of this generality, sentences are assigned a vast number of readings ("forced ambiguities") in addition to their legitimate readings.

These difficulties did not change with the advent of transformational recognizers [7, 9, 10, 14] in 1964. Due to the lack of comprehensive grammars and a complete set of transformational rules, these recognizers cannot be used for the analysis of arbitrary sentences in natural language. (Cf. [8]). It may also be significant that the advances in the theory of transformational grammar— the incorporation of a lexical component with semo-syntactic and semantic features, the introduction of output constraints, derivational constraints, and transderivational constraints— have not been incorporated into transformational recognition procedures.

The dissatisfaction with transformational grammar as described in *Aspects of the Theory of Syntax* [2] has led in the meantime to a schism among generative linguists, with the universal base hypothesis opposing an "extended" standard transformational grammar. Moreover, the general disaffection for the concept of a grammar as a device which generates individual sentences can be observed from various attempts to tackle the problem of producing or recognizing sentences in discourse by positing so-called text grammars.

We feel that the difficulties in the production of transformational grammars for language are mainly due to the unnecessary complexity of the transformational apparatus. A transformational grammar was, originally, a device which generates all and only the grammatical sentences of a (surface) language. The grammar supposedly generated deep structures from which— by means of transformations— well-formed surface strings were derived.

The advent of *Aspects* with its lexical component and embedding of sentences increased the power of the phrase-structure component; it was now able to generate well-formed and ill-formed sentences. The transformational component obtained an additional function, the "filtering function," whose purpose was

to delete all output strings which were not well-formed. These could be recognized from the occurrence of "non-surface" terminals: dummy symbols, and internal sentence boundary marks. That this filtering function did not suffice to eliminate all ill-formed strings has been shown. And, so far, the additional conditions stated above have had to be introduced in order to guarantee the well-formedness of the output string.

With this in mind the question naturally arises— Why not guarantee the well-formedness of the output string by means of an output grammar? It is certainly interesting, if not significant, that the centers which have made the most important advances in the analysis of sentences in natural language (the Transformation and Discourse Analysis Project at the University of Pennsylvania, and the CETA group in Grenoble) operate with a transformational apparatus but with a surface grammar. The addition of a surface grammar component has an obvious advantage in that the linguist is able to describe the strings of language in a manner which has been long familiar to him and which linguistic tradition has used for centuries. The transformational component can then be considerably simplified. In particular, the ordering of transformations which had originally been necessary to guarantee well-formedness of the output string can now be taken over by the surface grammar.

In conclusion — past experience has clearly demonstrated that, due to the large number of rules required, surface analysis by means of context-free grammars with simple symbols cannot be performed. Further, a grammar appropriate for surface analysis must permit the linguist to express the linguistic generalizations that he has been accustomed to make: that a sequence of constituents forms a constituent only if each constituent has the syntactic and semantic properties required for the well-formedness of the string.

---

In the remainder of this report we give a general outline of the linguistics research system (LRS), a grammatical model for the mechanical recognition and production of sentences in natural language used for machine translation purposes. A more comprehensive description is given in [5].

During this contract period, the theoretical group at LRC concentrated on disambiguation of sentences and on improving certain parts of the system to achieve greater economy in processing. Detailed descriptions can be found in [6]. The linguistic group was engaged in correcting and updating the existing German and English lexical data bases by assigning syntactic and semantic selection restrictions to lexical items. The systems group was concerned with a) reducing the size of existing LRS dictionaries

without loss of information while converting them to the new LRS format; b) constructing supporting programs to expedite and facilitate the updating of the LRS lexical data bases; and, c) constructing a part of the LRS grammar maintenance and LRS systems programs. Detailed descriptions of these activities can be found in Sections II, III, and IV.

# SECTION I

## *THEORY*

## THE LINGUISTICS RESEARCH SYSTEM

The purpose of the Linguistics Research System (LRS), which is being constructed under this contract at the Linguistics Research Center of the University of Texas at Austin, is to provide a description and explanation of human linguistic capabilities by performing recognition and production of sentences in natural language, in order to achieve mechanical translation. The LRS is a system of components which can be connected like building blocks to form larger configurations. Each component consists of a set of algorithms and instructions which are executed by the algorithms and which modify the general operations of the algorithms in a prescribed way. Such instructions are linguistic rules of various kinds: dictionary rules, syntactic rules, and interpretation rules, transformation rules, mapping rules, selection rules, rejection rules, and others.

The LRS is based on the following linguistic assumptions:

1) that grammatical relations can be more easily and correctly stated for so-called standard strings than for surface strings;

2) that surface information is necessary for correct semantic interpretation;

3) that synonymous sentences can be reduced to the same "universal" representation.

In its basic configuration the LRS is a grammatical model for the recognition and production of synonymous surface sentences with identical or different deep structures. By deep structures we mean the stage of a sentence derivation in standard transformational grammar when all base component rules, constituent and feature rewriting rules, have applied but before lexical insertions have been performed.

### 1.1 Canonical Forms

The purpose of this model is to associate with each sentence in a natural language all its semantic readings or canonical forms (KF), and to derive from a given KF t all sentences with the semantic reading t. A sentence which has n distinct semantic readings has n distinct KF's. Two different sentences t and u

which have one semantic reading in common have one KF in common. Sentences of different languages which are translations of one another have at least one KF in common.

A canonical form consists of a sequence of connected simple KF expressions. K, the language of KF's, has the following properties:

a) Each simple KF expression is a primitive element of K (i.e., it has one and only one [atomic] semantic interpretation). If a surface terminal q has n different senses, then n different KF expressions (simple or connected) represent the different senses of q.

b) No two different KF expressions p and q are synonymous. If two surface terminals have one sense in common, then that reading is represented by the same KF expression.

## 1.2 Normal Forms

Because of the difficulties involved in the construction of KF's, LRS represents the meaning of sentences by means of normal forms (NF).

The NF's of a language are distinct from the KF's in that NF lexical primitives may represent either atomic (simple) or molecular (connected) KF expressions. Thus the NF primitive, $bachelor_1$, corresponds to the connection of the four simple KF expressions *unmarried·human·adult·male*.

## 1.3 Mechanical Translation

The process of deriving from a surface sentence t all the NF's of t is performed by the following components:

the surface component

the standard component

the normal form component.

The surface component assigns to each surface sentence t all its syntactic readings according to the surface grammar, and derives from those a tentative standard string by means of the transformation instructions contained in the rules which apply to t. Tentative standard strings consist of complex standard terminal symbols. These are surface terminals with their (possibly disambiguated) dictionary interpretation, and dummy symbols which were introduced by the transformations. Dummy symbols represent grammatical morphemes and elided lexical items. Elements which were discontinuous in the surface are contiguous in the

tentative standard strings.

The standard component then analyzes these strings with the standard grammar which assigns a standard description to all well-formed strings, and filters out all ill-formed strings.

The NF component finally interprets the readings of the remaining standard strings by means of the NF grammar which assigns NF expressions to individual or connected standard subtrees.

Production, the reversal of the recognition process, is also performed in three steps.

a) The normal form component— by means of the NF grammar of the output language— derives from the NF reading of the input sentence, which is identical to the NF reading of the output language, all the associated tentative standard readings of the output string t.

b) The standard component— by means of the conditions and operations stated in the standard grammar rules of the output language— selects all well-formed standard readings from the tentative standard readings and filters out all ill-formed readings. The standard component then associates with each standard reading the corresponding tentative surface strings.

c) The surface component— by means of the rearrangement grammar of the output language— then assigns a surface description to all well-formed surface strings and filters out all ill-formed surface strings, i.e., those which are either not accepted or do not meet the output conditions of the rearrangement grammar. The transformation instructions associated with the rearrangement rules finally delete the standard dummy symbols, reintroduce lexical pieces which had been deleted after surface analysis, and rearrange the remaining terminals in surface word order.

## 1.4 Subscript Grammars

Four grammars— surface, standard, normal form, and rearrangement— exist for each language. The non-terminal and terminal vocabulary symbols of each grammar are complex symbols, except for the terminal symbols of the surface grammar. Each complex symbol consists of a category symbol and zero or more subscript or feature symbols; each subscript may have zero or more values.

The grammar rules used during the recognition and production of sentences (*both* of which are performed as a bottom-to-top direct substitution analysis), are generated by the processing

algorithms by means of instructions represented as context-free rule schemata. A rule schema successfully analyzes a string of vocabulary symbols if each rule constituent is non-distinct from the symbol it analyzes, and if all the relations stated between rule constituents in the rule schema hold for the corresponding analyzed symbols.

If a rule schema is successfully applied, a new vocabulary symbol is constructed according to the instructions stated in the antecedent of the rule schema.

The conditions that may be stated for individual constituents in a rule consequent are:

a) A particular category symbol either may not or must contain a particular subscript or combination of subscripts;

b) A particular subscript symbol may not or must contain a particular value or combination of values;

c) Operations between subscripts of different constituents may not or must be successful. (These operations, the set-theoretical operations Intersection, Sum, and Difference, are performed with the values of the specified subscripts.)

The advantages of a subscript grammar are numerous. It permits the expression of relations such as agreement and government which correspond to the intuition of the human speaker. Similarly, grammatical, semantic, and stylistic categories can be conveniently expressed.

## 1.5 Syntactic Grammars

Each rule schema of each grammar consists of a syntactic part and an optional transformational part. For surface and standard grammar, the syntactic part of each rule schema consists of context-free rewrite rules. The transformational part contains only transformations whose structural description is satisfied by a string of symbols interpreted by the constituents of the rule schema consequent. The transformations possible in surface and standard grammar are permutations, deletions, and insertions. The transformations are "feature sensitive": in particular, it is possible to lexicalize features of a constituent and to "featurize" terminal or non-terminal constituents. Thus, words like *up* which form a lexical unit with some verbs, e.g., *look something up*, can be assigned as a feature to the head of the verbal construction, resulting in *look something*.
*+up*

## 1.6 Normal Form Grammar

The rules of the NF Grammar differ from surface and standard rules in two respects:

a) They apply to connected trees;

b) They are not rewrite rules.

An NF rule applies to all trees (terminal, non-terminal, or combinations of them) whose nodes, labeled by complex symbols, are non-distinct from the complex symbols in the consequent of the NF rule. The antecedent of the NF rule assigns a particular semantic reading, an NF expression represented by that antecedent, to all trees to which it applies. Since NF expressions apply to trees whose nodes are labeled by complex symbols, it is possible to assign a particular NF reading to a terminal k with a particular part-of-speech interpretation and with a particular selection restriction. At the same time, all trees $t_1$, $t_2$,...$t_n$ interpreted by the same NF expression k are substitutable for one another, regardless of whether the root and end nodes of tree $t_i$ are identical or different from those of tree $t_j$.

It is thus possible to define synonymy relations between words of different part-of-speech and between different syntactic structures and terminal structures (e.g., lexical units and idiomatic expressions; lexical units and phrasal expressions; and, lexical units which have an internal variable slot), without affecting their transformational possibilities. Examples of such paraphrases can be found in [5], pp. T217-68.

## 1.7 Analysis Procedure

The recognition and production of strings is performed as a bottom-to-top analysis. We believe that analysis procedures like those of Earley [3] or those based on state-transition diagrams [1, 9, 14] do not operate as efficiently with LRS grammars due to the complexity of their symbols and the large number of permutations of constituents typical of highly inflected languages such as German.

We selected bottom-to-top analysis for the reasons which follow.

a) It permits an easier treatment of ill-formed strings (k-strings) within well-formed strings which occur frequently in translations, e.g., formulas, foreign names, foreign citations, etc..

b) It permits the adding of new syntactic or semo-syntactic values to the lexicon without a concurrent change of the non-terminal grammar rules. Assume, for example, that one discovers a sub-class of adjectives which modify only a certain type of nouns. The addition of the new semantic feature under the subscript "type" only requires changing the dictionary rules for the nouns and adjectives affected.

None of the word formation rules or syntactic rules will need to be changed. This advantage would be lost in a top-to bottom analysis where, in addition to the dictionary rules, the tables for the subscript "type" for nouns and for adjectives would have to be changed.

c) Finally, tree structures which interpret ambiguous strings can be conflated to a single tree structure if all labels of the tree nodes have the same category symbol. The number of intermediate analyses, similar to state transition diagrams, is thus considerably reduced. A similar conflation occurs in the representation of the normal forms of sentences which contain semantically ambiguous items.

The economy of this analysis procedure was further increased by the introduction of:

a) left-context-sensitive dictionary analysis (cf. 3.4.1);

b) intermediate choice algorithms which— based on well-formedness conditions— destroy all inappropriate readings after dictionary and word analysis; and,

c) context-sensitive rejection rules, which apply during word analysis and whose instructions are executed during word choice. Word Choice tags all those nodes on which no syntactic rule may build within the analyzed text.

## 1.8 Intra-Sentential Disambiguation

The most powerful feature is the system's capability of performing semantic disambiguation of lexical items in context after sentence analysis without having to trace down the tree branches from the node S. This is made possible by means of trace operators which are associated with the disambiguating values of ambiguous lexical items. These operators cause the system to remember the location of these lexical items and to disambiguate them only if a disambiguating environment is given.

## 1.9 Changes in Subscript Grammar Format and Storage

Certain modifications in the format of writing and storing subscript rules were made during the reporting year. The most significant are:

1) the now-permissible separation of condition and operation statements in subscript rules, and,

2) the method of storing the grammar for actual analysis.

### 1.9.1 New Format for Operation Statements

The encoding scheme below was introduced in order to eliminate the ambiguity resulting from two or more linked operations. For example, consider the rule:

```
                ①          ②          ③          ④
C 5   V NP   =   V DET      V A        V N
                 - 3.1GD    . 4.1GD    $ GD
```

(The encircled digits identify the rule fields.) Under the old convention it is not obvious whether the operation in field 2 (i.e., -3.1GD) means: perform the difference operation between—

a) the value set of the workspace subscript GD for DET and the value set of the workspace subscript GD for A,

or

b) the value set of the workspace subscript GD for DET and the value set resulting from the intersection indicated at 3.1.

In the first, a), the operations at 2.1 and 3.1 are disjoint and can be done in any order. In the second, b), the operation stated in 3.1 must be done first.

The operation statement for a subscript may now be separated from its condition statement. Rule 12, which was originally encoded as

```
C 12  V NO   =  V A        V N
                . 3.1GD    $ GD
```

or

```
C 12  V NO   =  V A        V N
                $ GD       . 2.1GD
```

may now be encoded as

```
C 12  V NO   =  V A        V N
                $ GD       $ GD
                . 2.1,3.1
```

The statement ". 2.1,3.1" represents "perform an intersection between the value sets of the subscript names enumerated at 2.1 and 3.1", i.e., GD of A and N.

Since the system treats a separated operation statement as if it were also a subscript, sequences of linked operations can be stated in a straightforward manner. Thus, for example, reading b) in Rule 5, above, can now be represented as—

```
C 5   V NP   =  V DET      V A        V N
                $ GD       $ GD       $ GD
                - 2.1,3.2  . 3.1,4.1
```

whereas reading a) is represented as—

```
C 5   V NP   =   V DET        V A        V N
                  $ GD         $ GD       $ GD
                  - 2.1,3.1    .3.1,4.1
```

No condition is imposed on the position where separated operation statements may occur. It is thus possible to place them in the most advantageous position from a processing point of view, i.e., the left-most constituent in a rule consequent, as for version b) at the bottom of the preceding page—

```
C 5   V NP   =   V DET        V A      V NP
                  $ GD         $ GD     $ GD
                  . 3.1,4.1
                  - 2.1,2.2
```

### 1.9.2 Storage of Analysis Grammars

The manner in which the word and syntax grammars are stored has a great influence on the speed of analysis. After investigating how the word and syntax algorithms would operate, a storage structure using a reverse columnar approach was chosen. The grammar in question is stored by columns, the first column containing all the unique last terms of rule consequents. The succeeding columns contain the penultimate terms, the antepenultimate terms, etc.. Associated with each term is a list of rule numbers in which it occurs. Each terminal, i.e., the left-most term of a rule consequent, is marked and has a pointer to its antecedent term.

The analysis programs construct the actual rule by means of the analyzed individual terms and their associated rule numbers. Since each unique nth rule term is stored only once, the method of storing the grammar as described above should facilitate the analysis as well as use a minimum of storage. As the grammar might exceed available core memory space, the storage method also ensures that most or all of the grammar that the analysis program needs at one time is kept in memory. If last terms are being analyzed for instance, all the last terms can be in memory; if penultimate terms are being analyzed, all the penultimate terms can be in memory, etc. We anticipate that this method of storing the grammar will result in a considerable increase in processing speed.

## 1.10 Example for Intra-Sentential Disambiguation

The capabilities of LRS for performing intra-sentence disambiguation may be shown by the analysis and standardization of the English sentence

*The page slept*

In this sentence, the noun *page* is ambiguous; one of its semantic

readings is the reading *BOY*, another the reading *PAGINA*. This ambiguity is represented in the dictionary rule 2 below, which applies to the noun *page*, by the subscript TY (for type) with the values HU and IN for HUman and INanimate. This ambiguity is resolved in the context of the verb *slept*, which requires an animate subject, indicated by the subscript TS with the value AN in rule 6. During the analysis of this sentence, the rule schemata apply in the order indicated by their numbers.

English dictionary and grammar rules:

```
1    V DET              =   * THE
     + NU(S,P)

2    V N                =   * PAGE
     + TY(HU,IN)
     + CL(05)
     T 1.1

3    V N                =   V N
     $ 2.1(X+AN+PO)         $ TY(*AN+*PO+HU)
     Λ 2

4    V N                =   V N
     + NU(S)                $ CL(01,..05)
     Λ 2

5    V NP               =   V DET          V N
     + PS(3)                .  3.1NU        $ NU
     $*2.1
     Λ 3

     CHOICE
     S m                    (m = 2-1)

6    V V                =   * SLEPT
     + CL(15)
     + OB(0)
     + TS(AN)

6    V V                =   * SLEEP
     + CL(07)
     + OB(0)
     + TS(AN)

7    V VP               =   V V
     + TN(PA)               $ CL(...15)
     + PS(1,2,3)
     + NU(S,P)
     Λ 2
```

```
8   V S      =  V NP       V VP      |  D #    D AUX     D #
   $ 3.3         . 3.1NU    $ NU      |         $ 3.5
   $*2.3         . 3.2PS    $ PS      |         $ 3.6
                 . 3.4TY    $ OB(*0)  |         $*2.1
                            $ TS      |         $*2.2
                            $ TN      |
                            $ VC(A)   |

   CHOICE
   A 1.1(3.3)
   A 1.2(2.3,3.4)
   S n                      (n = 2-4-1-3-5)
```

Rule 1 assigns the word *the* the interpretation DETerminer and states that its NUmber is Singular or Plural.

Rule 2 assigns the word *page* the interpretation Noun of the paradigmatic CLass 05 and the values HUman and INanimate of the subscript TY. The subscript T 1.1 indicates that the values in the first subscript of the first rule term, in this case of TY, represent semantic ambiguity. The effect of the T operator is that the address of this subscript, given in brackets in the tree diagram below, is associated with the subscript TY.

Rule 3 is a redundancy rule which states that all nouns with the value HUman which have neither the value ANimate nor the value Physical Object add the values ANimate and Physical Object. The expression Λ 2 in the antecedent is an instruction to the algorithm to carry along all the subscripts of the second constituent not mentioned in the second rule term.

Rule 4 states that nouns of particular paradigmatic CLasses, if followed by zero ending, become nouns with the NUmber Singular. Again, Λ 2 results in the carrying along of the non-mentioned subscripts.

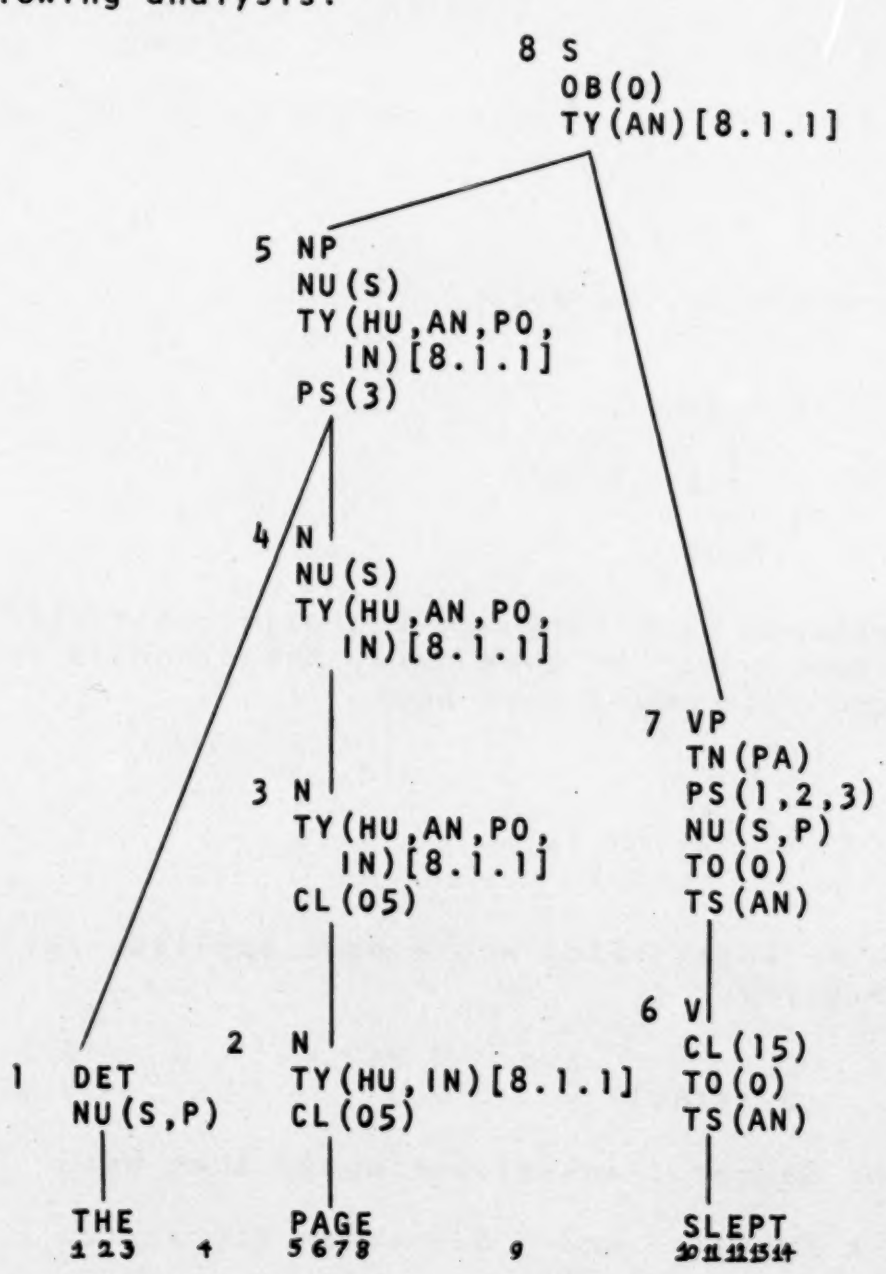
Rule 5 states that the sequence of DETerminer and Noun results in a Noun Phrase, provided that the DETerminer and the Noun agree in NUmber. The instruction . 3.1NU is to be read as "intersect the values of the subscript NU with the values of the first subscript of the constituent matched by the third rule term." The Noun Phrase is assigned the feature "third person" and the NUmber in which the two terms agree; the non-mentioned subscripts of term 3 are carried along.

Rule 6 assigns the word *slept* the reading Verb of paradigmatic CLass 15. OB(0) stands for "requires zero object," TS(AN) stands for "the subject must be animate." As we see in the next rule, allomorphs of a morpheme are assigned the same rule number. They have in common all subscripts except for the subscript which indicates paradigmatic CLass.

Rule 7 rewrites all Verbs of CLass 15 as full VerBs in the PAst TeNse, in the first, second or third PerSon and in the NUmber Singular or Plural. Λ 2 results in the carrying along of all features of the underlying verb.

The syntactic part of rule 8 consists of the first three terms which rewrite a Noun Phrase followed by a Verb Phrase as a Sentence provided the Noun Phrase and the Verb Phrase agree in NUmber and PerSon and provided that the TYpe of the Noun Phrase has a value in common with the subscript TS of the Verb Phrase. In addition, the verb phrase must dominate an intransitive verb (objects of transitive verbs are dominated by S not by VP). These subscripts are artificially associated with S to permit an easier execution of the rule's choice statement.

The application of these rules to the input sentence results in the following analysis:




*Note that "space" occurs as the 4th and 9th text symbols.*

After syntactic analysis, the choice statements in rule 8 are executed. A 1.1(3.3) reads "take the value of the first subscript in field 1 and weight it in the address associated with the third subscript in field 3 if there is such an address." Thus only the instruction A 1.2(2.3) of A 1.2(2.3,3.4) is executed. Syntactic choice also introduces the dummy terms of rule 8 and assigns the order 2-4-1-3-5 to the terms and dummy terms in the rule consequent.

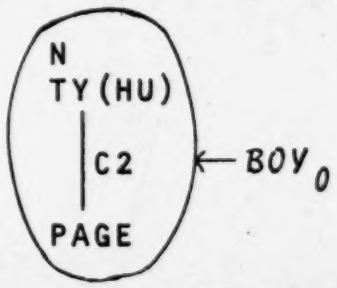
The standardization program then derives the following disambiguated string, where the noun *page* no longer has the features "human or inanimate" but only "human," as indicated by the subscript and value TY(HU) below.

```
D #    C 2          C 1          D AUX        C 6          D #
         N            DET        $ PS(3)        V
         TY(HU)       NU(S,P)    $ NU(S)        CL(15)
         CL(05)                  $ TN(PA)       TO(0)
                                                TS(AN)
```

The noun is assigned the interpretation *BOY* by the NF rule

```
V BOY      ≡     C 2
D 0              $ TY(HU)
```

which we can represent by the graph




Had the input sentence been *They saw the page*, no disambiguation would have been possible. In that case, the standard representation of the noun *page* would have been

```
C 2
  N
  TY(HU,IN)
  CL(05)
```

to which the two NF rules below would have applied, reflecting the semantic ambiguity.

```
V BOY      ≡     C 2          and    V PAGINA     ≡     C 2
D 0              $ TY(HU)           D 0                $ TY(IN)
```

The two resulting German translations would then be:

*Sie sahen den Knaben* and *Sie sahen die Seite.*

# SECTION II

## *LEXICOGRAPHY*

### 2.1 Existing Data

The German and English lexicographic data which were available at the beginning of the reporting period included the German monolingual machine-processable dictionary, two English monolingual machine-processable dictionaries, a German verb list, an English verb list, and German-English past participle and noun lists.

#### 2.1.1 The German Dictionary

The German dictionary consists of approximately 40,000 entries. Since stem variants of nouns, adjectives, and verbs constitute separate entries, these 40,000 dictionary entries represent approximately 35,000 German word stems. Each entry is classified as belonging to one of the following categories: noun, adjective, verb, adverb, determiner, pronoun, preposition, conjunction, or separable verbal prefix. In addition to these categories, paradigmatic features are assigned to nouns, adjectives, and verbs. Nouns have a feature, "gender," which identifies them as masculine, feminine, neuter, or (in the case of *pluralia tantum* nouns) plural. Adverbs which may be used to modify nouns are marked with respect to their position relative to the modified noun phrase: preposed (e.g., *sogar die Roemer*), or postposed (e.g., *dieser Satz hier*).

#### 2.1.2 German Lexical Lists

In order to expand the LRC machine processable dictionaries, lists of German verbs and of past participles commonly used as adjectives had previously been compiled, and compilation of a list of German nouns had begun. All information was coded from the *German-English English-German Dictionary* by Wildhagen and Heraucourt since this dictionary contains a comparatively large amount of explicit syntactic and semo-syntactic information.

##### 2.1.2.1 The German Verb List

The German verb list originally contained approximately 18,400 entries. To this list, all stem variants of irregular verbs were added automatically, resulting in a total of approxmately 30,000 entries. The entries contain a large amount of

syntactic but only a modicum of semo-syntactic information (approximately for 12% of all verb entries).

a) Prefixes precede the verb stem. Separable prefixes are followed by a blank space, inseparable prefixes by a hyphen and a blank space. Examples:

AUF STEH (separable prefix)

VER- ZWEIFEL (inseparable prefix)

Note that the infinitive endings are stripped from the stem. All stem variants of irregular verbs are entered in the dictionary with identical semo-syntactic and syntactic information, but with different paradigmatic information.

b) Transitivity. Each verb in the verb list is identified by a descriptor as transitive (VT), intransitive (VI), or reflexive (VR).

c) Case government is indicated for all transitive verbs as genitive (GEN), dative (DAT), or accusative (VT or VR). Descriptors indicating case government may also contain information about the semantic type of the object:

JDN = human, accusative

JDM = human, dative

JDS = human, genitive

ETW = non-human, accusative

ETW DAT = non-human, dative

ETW GEN = non-human, genitive

VR DAT = reflexive, dative

E-A = reciprocal (*einander*)

ES = object must be *es*

d) Verbs which govern prepositional objects are marked by the specific preposition(s) they may take. Those prepositions which govern either dative or accusative are distinguished by case descriptors:

AN ACC = *an* with accusative

AN DAT = *an* with dative

Prepositions are followed by descriptors specifying the semantic type of object required (as in AUF JDN), or by SICH (if the prepositional object must be reflexive), whenever this information was recognized in Wildhagen.

e) The semantic type of subject required by a verb (if

indicated) is shown by (P) for human, (T) for animal, or (S) for inanimate.

f) The auxiliary taken by a verb in perfect tense forms is indicated as follows:

takes *sein* = S

takes either *haben* or *sein* = S H

takes *haben* = unmarked

### 2.1.2.2 The German-English Noun List

Work on the compilation of a list of German nouns had been in progress for some time. The information coded includes gender, number (for *pluralia* and *singularia tantum* nouns), case government (including prepositions) for deverbative nouns, and English translation equivalents. Whenever information was given in Wildhagen about the area of discourse to which a particular translation is restricted, this information was coded with the proper translations.

### 2.1.2.3 The German-English Past Participle List

The list of German past participles consists of approximately 1,100 entries. It contains primarily those past participles which are frequently used as adjectives and whose meaning and translation cannot be automatically derived from the underlying verb stem, e.g., *aufgebracht* or *interessiert*, or the English adjective *excited*. [Note the difference in meaning between the past participial and the adjectival usage:

*The electron was excited.* (passive)

*The man was excited.* (active)]

Also included are past participles whose stem does not function as a verb in modern German, as for example, *bestuerzt (aghast)*, or *betagt (aged)*. The descriptors coded with these entries indicate case government (including prepositions), semantic type of object required, and English translation equivalents.

### 2.1.3 The English Dictionaries

The English lexicographic data base existing at the beginning of this reporting period consisted of two monolingual machine-processable dictionaries: a) the so-called WEBSTER dictionary, based on *Webster's New Collegiate Dictionary*, which contains approximately 77,500 entries (47,300 nouns, 20,100 adjectives, 9,200 verbs, plus adverbs and function words), and

b) the so-called LRMD, which was derived from the Russian Master Dictionary (RMD) and contains approximately 47,300 entries (34,000 nouns, 7,800 adjectives, 4,800 verbs, plus adverbs and function words). Each word stem is assigned to one of the categories: noun, adjective, verb, adverb, determiner, pronoun, preposition, or conjunction. In addition, nouns, adjectives, and verbs are assigned to paradigmatic classes and have a feature indicating vocalic or consonantal onset. Nouns in the LRMD are also subclassified as human or non-human. A small set of adjectives has a subscript identifying them as possible post-nominal modifiers, e.g., *afire*.

### 2.1.4 The English Verb List

The English verb list was compiled from *The Advanced Learner's Dictionary of Current English* by Hornby, Gatenby, and Wakefield, and contained approximately 6,400 entries. The syntactic information given for list entries included the permissible types of complementation for each verb: objects (direct and indirect, either of which may be in the form of a prepositional phrase), predicative complements, adjectives, adverbials, infinitives (unmarked and marked by *to*), present and past participles, *that*-clauses, interrogative clauses, gerunds, and combinations of these.

## 2.2 Progress

The lexicographic work done during the reporting period consisted of the revision of the existing lexical lists, the addition of translation equivalents, and the development of a general system of syntactic and semo-syntactic features to be used in the further subclassification of lexical elements.

### 2.2.1 Purpose

The purpose of the lexicographic work performed at the Center is manifold:

a) to make the LRC machine-processable dictionaries as comprehensive as possible in order to provide for maximum recognition of lexical elements in input texts and for all necessary translation equivalents;

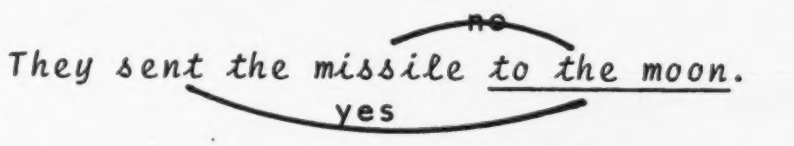
b) to prevent ambiguous readings of phrases and sentences by means of lexical information;

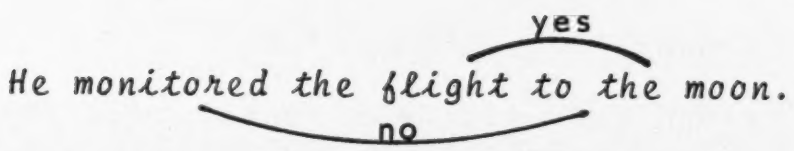
c) to permit the selection (on the basis of lexical features) of the proper translation equivalent for a lexical item

from two or more translation equivalents;

d) to guarantee production of well-formed sentences only.

The first of these points is obvious. The following examples may illustrate point b):




For both examples (as for each surface sentence in which a prepositional phrase immediately follows a noun phrase in post-predicate position), there are two possible analyses. One analyzes the adverbial as a post-nominal modifier (represented by an arrow above the sentence in the examples given). The other analyzes it as a verb modifier (arrows below the sentences). Given the necessary syntactic or semo-syntactic information, such sentences can often be disambiguated. In this instance, the relevant distinction lies in whether the verb and noun may be modified by an adverb of direction. (The correct analyses of the examples above are indicated.)

An example for point c), the need for selection of proper translations based on lexical information, is the German verb *abfuettern*. It has two possible English translations: *feed* if the object is animate, *line* if the object is inanimate (more precisely, articles of clothing). The choice between these two translation equivalents can easily be made if the distinction between animate and inanimate objects is made in the verb dictionary, and if nouns are sub-classified accordingly.

Point d), the production of well-formed sentences, is clear: English verbs require certain syntactic patterns and may not be used in other patterns. For example, the German sentence

*Sie erklaerten ihm das Problem.*

must be translated as

*They explained the problem to him.*

but not

**They explained him the problem.*

while the sentence

*Wir gaben diesem Phaenomen einen neuen Namen.*

may be translated as

*We gave a new name to this phenomenon.*

or *We gave this phenomenon a new name.*

The information coded in the various lexical lists will later be added to the German and English machine-processable dictionaries of the Center.

### 2.2.2 Work Done: German

#### 2.2.2.1 The German-English Noun List

Compilation of the German noun list was continued. For each German entry we coded English translation equivalents and any relevant features indicating gender, number (for *tantum* nouns), case government (including prepositions) for deverbative nouns, and area of discourse or stylistic level (e.g., <TECH>, <MED>, <PHYS>, etc.). This work progressed through the German noun *Exzess*, reaching a total of approximately 20,000 German nouns.

#### 2.2.2.2 Revision of the German Verb List

The German verb list was revised in its entirety. This revision included the following:

a) correction of miscoded or mispunched key-words or descriptors, and addition of missing entries;

b) addition of case information to all German prepositions which may be used with either dative or accusative;

c) addition of the descriptors ZI (*zu-Infinitiv*, i.e., marked infinitive) and DASS (*that*-clause) to those German verb entries which take one or both of these verb complements;

d) introduction of the symbol + between verb complements which may be used as double objects with the particular verb.

#### 2.2.2.3 Addition of the Translation Equivalents

The English translation equivalents given in Wildhagen for German verbs were added to the revised German verb list. In the process of this work, German verb entries were split into more than one entry whenever different English translations for a German verb could be associated with specific groups of German features.

Examples:

VER- MESS VT = MEASURE (EO LAND)

VER- MESS VR = MEASURE INCORRECTLY

VER- MESS VR + ETW GEN ZI = DARE, VENTURE

as in:

*Sie vermassen diese Gegend.*

*They measured this area.*

*Dabei hatten sie sich vermessen.*

*They have measured incorrectly in this case.*

*Sie vermessen sich, diese Vermutung als Tatsachen hinzustellen.*

*They dare to represent these assumptions as facts.*

Additional information which was given in Wildhagen for the purpose of selecting proper translation equivalents was coded with each English translation to which it pertained. This type of information consists of:

a) the area of discourse in which a particular translation would be used (given in the list in angled brackets, e.g., <PHYS>, <MED>, etc.); or,

b) selection restrictions in the form of particular nouns given as sample subjects or objects of the German verb or of its English translation. These were added to the translations and were marked as English or German, subject or object, by two preceding letters: ES (English subject), EO, GS, or GO.

In addition, some English translation equivalents in Wildhagen are accompanied by syntactic or semo-syntactic information. Such data was incorporated in the noun list in the form of four descriptors:

AP = a person (human object)

AP'S = a person's (human possessive pronoun)

ATH = a thing (inanimate object)

OS = oneself (reflexive object)

Finally, verb entries which are used in Wildhagen in a verb phrase with the German verb *lassen* (*let, have,* as in *have someone do something*)were marked in this bilingual verb list. This information will be used in future studies of verb phrases of this type.

### 2.2.3 Work Done: English

At the beginning of this contract period, the English verb

list (EVL) consisted of 6,547 entries which had been copied from *The Advanced Learner's Dictionary of Current English* by Hornby, Gatenby, and Wakefield. This is, to our knowledge, the only dictionary which indicates for verbs the object complement and adverbial complement environment in which the verb may occur.

Apart from its value as a tool for linguistic analysis, the EVL was created for two reasons: to guarantee the production of well-formed English sentences, and to be able to associate with a particular verb the syntactic pattern in which the verb can be used with a given meaning.

2.2.3.1 Classification in the Hornby Dictionary

In addition to the classification indicated by the patterns below, verbs are redundantly marked as transitive or intransitive if this is applicable. Verbs which require a reflexive object are marked as VR; modals and auxiliaries, as "anomalous finites".

**VERB PATTERNS**

P1. Verb + Direct Object
*He cut his finger.*

P2. Verb + (not) to + Infinitive
*He intended to go.*

P3. Verb + Noun or Pronoun + (not) to + Infinitive
*I told the servant to open the window.*

P4. Verb + Noun or Pronoun + (to be) + Complement
*We proved him (to be) wrong.*

P5. Verb + Noun or Pronoun + Infinitive
*They felt the house shake*

P6. Verb + Noun or Pronoun + Present Participle
*They left me standing outside.*

P7. Verb + Object + Adjective (object complement)
*The sun keeps us warm.*

P8. Verb + Object + Noun (object complement)
*They named their son Henry.*

P9. Verb + Object + Past Participle
*She had a new dress made.*

P10. Verb + Object + Adverbial Adjunct

*Put it here.*

P11. Verb + *that*-Clause

*He explained that nothing could be done.*

P12. Verb + Noun or Pronoun + *that*-Clause

*We satisfied ourselves that the plan would work.*

P13. Verb + Interrogative Adverb (except *why*) + *to* + Infinitive

*He is learning how to swim.*

P14. Verb + Noun or Pronoun + Interrogative Adverb (except *why*) + *to* + Infinitive

*The patterns show you how to make sentences.*

P15. Verb + Interrogative Adverb + Clause

*I don't mind where we go.*

P16. Verb + Noun or Pronoun + Interrogative Adverb + Clause

*They asked us when we would be back.*

P17. Verb + Gerund

Group A - replacing the gerund with an infinitive results in a change of meaning.

*We stopped talking.*

*We stopped to talk.*

Group B - the gerund may be replaced by an infinitive without a change of meaning.

*He began talking.*

*He began to talk.*

Group C - the gerund is equivalent to a passive infinitive.

*That needs explaining.*

*That needs to be explained.*

P18. Verb + Direct Object + Indirect Object

Group A - the indirect object is preceded by the preposition *to* and may occur without a preposition before the direct object.

*Throw that ball to me.*

*Throw me that ball.*

Group B - the indirect object is preceded by the preposition *for* and may occur without a preposition before the direct object.

*Have you left any for your sister?*

*Have you left your sister any?*

Group C - covers all direct object + indirect object constructions other than those stated in Groups A and B.

*I explained the difficulty to him.*

P19. Verb + Indirect Object + Direct Object

Group A - are those verbs which can be used with the preposition *to* in Pattern 18A.

*He handed me the book.*

*He handed the book to me.*

Group B - are those verbs which can be used with the preposition *for* in Pattern 18B.

*Buy me one.*

*Buy one for me.*

Group C - are those verbs which are rarely or never used in Pattern 18.

*I struck him a heavy blow.*

P20. Verb + (*for*) + Complement of duration, distance, price or weight

*The rain lasted (for) a whole week.*

*It cost ten dollars.*

P21. Verb alone

These are intransitive verbs. Some verbs which are normally used with an object may also be used in this pattern, the object being understood.

*Fire burns.*

*The moon rose.*

P22. Verb + Predicative Complement

*This is a boat.*

P23. Verb + Adverbial Adjunct

*We must turn back.*

P24. Verb + Preposition + Object

The verb and preposition combine to form a new transitive verb followed by an object which can be a noun, pronoun, gerund, phrase or clause.

*Look at the blackboard.*

*He called on me.*

P25. Verb + *to* + Infinitive

Group A - the infinitive is one of purpose or aim.

*I went to buy some books.*

Group B - the infinitive indicates result or outcome.

*How can I get to know her?*

Group C - the infinitive is equivalent to a co-ordinate clause.

*He awoke to find the house on fire.*

Group D - the infinitive is the main verb.

*I chanced to meet him in the park.*

Group E - the infinitive is used after finites of *be* for a variety of meanings.

*Nobody is to know.*

*This I was to learn later.*

Group F - contains as the only member the verb *going to*:

*He is going to walk home.*

2.2.3.2 Frequency of Patterns in EVL, 1969

The entries in EVL were subjected to a glossary run. The results are represented in Table I, which follows.

TABLE I: Frequency of Patterns in EVL

| Pattern | Frequency in EVL |
|---|---|
| P1 | 4248 |
| P2 | 80 |
| P3 | 120 |
| P4 | 59 |
| P5 | 14 |
| P6 | 17 |
| P7 | 96 |
| P8 | 24 |
| P9 | 7 |
| P10 | 1670 |
| P10B | .1 |
| P11 | 181 |
| P12 | 26 |
| P13 | 42 |
| P14 | .8 |
| P15 | 61 |
| P16 | 10 |
| P17 | 14 |
| P17A | 55 |
| P17B | 16 |
| P17C | .3 |
| P18 | 910 |
| P18A | 17 |
| P18B | 80 |
| P18C | .8 |
| P19 | 76 |
| P19A | 16 |
| P19B | .9 |
| P19C | 10 |
| P20 | 78 |
| P21 | 2074 |
| P22 | 45 |
| P22D | .1 |
| P23 | 1372 |
| P24 | 1121 |
| P24A | .1 |
| P24B | .1 |
| P25 | 139 |
| P25A | .1 |
| P25B | .1 |
| P29* | .312 |

*P29 refers to verbs which were not classified in the Hornby Dictionary.

2.2.3.3 Subsequent Work

The purpose of the work performed during the first year of the contract period was to improve EVL by making the original classification scheme more precise, and to add to it the same semo-syntactic selection restrictions as those of the German verb list.

Thus, two of the Hornby verb patterns, P10 and P23, were redefined. Pattern 10, for which Hornby gives as examples

*He brought his brother to see me.*

*They treat their sister as if she were only a servant.*

was restricted to

Verb + Object + Movable Adverbial Particle

*He took off his hat.*

*He took his hat off.*

Similarly, Pattern 23 was defined as

Verb + Adverbial Particle

*Get up.*

*Sit down.*

The actual updating of EVL involved:

a) the addition of the adverbial particles with which each verb in the new P10 and P23 could occur;

b) the addition of the preposition(s) which each verb in P24 (Verb + Prepositional Object) required;

c) the subclassification of the verbs in the general classes P17, P18, and P19 into the corresponding subclasses A, B, and C shown above in 2.2.3.1;

d) the specific classification of all verbs which had, as a stop-gap measure, been assembled under P29; and

e) the addition of the descriptors H, N, M, K, I (for: human, non-human animate, non-animate, non-animate concrete, non-animate abstract, respectively) to all patterns in which a noun phrase object complement occurred.

This updating process resulted in a new EVL, which consists of 10,431 entries. Comparison of frequency of descriptors in the new and the original EVL is made in Table II, which follows.

TABLE II: Frequency of Patterns in New and Original EVL

| Pattern | new EVL | Original EVL |
|---|---|---|
| P1 | 4267 | 4248 |
| P2 | 79 | 80 |
| P3 | 122 | 120 |
| P4 | 59 | 59 |
| P5 | 14 | 14 |
| P6 | 17 | 17 |
| P7 | 92 | 96 |
| P8 | 22 | 24 |
| P9 | 7 | 7 |
| P10 | 1269 | 1670 |
| P10B | - | 1 |
| P11 | 179 | 181 |
| P12 | 27 | 26 |
| P13 | 41 | 42 |
| P14 | 7 | 8 |
| P15 | 70 | 61 |
| P16 | 10 | 10 |
| P17 | 15 | 14 |
| P17A | 80 | 55 |
| P17B | 16 | 16 |
| P17C | 3 | 3 |
| P18 | - | 910 |
| P18A | 63 | 17 |
| P18B | 32 | 80 |
| P18C | 1743 | 8 |
| P19 | - | 76 |
| P19A | 58 | 16 |
| P19B | 30 | 9 |
| P19C | 14 | 10 |
| P20 | 76 | 78 |
| P21 | 2166 | 2074 |
| P22 | 42 | 45 |
| P22D | - | 1 |
| P23 | 866 | 1372 |
| P24 | 1778 | 1121 |
| P24A | - | 1 |
| P24B | - | 1 |
| P125 | 139 | 139 |
| P25A | 1 | 1 |
| P25B | 1 | 1 |
| P29 | - | 312 |

The complete list of entries in the new EVL, subdivided as follows, is attached to the report, *Normalization of Natural Language for Information Retrieval* by Lehmann and Stachowitz.

a) Verbs which are both transitive and intransitive
   1) consisting of more than one word
   2) consisting of one word only

b) Verbs which are transitive only
   1) consisting of more than one word
   2) consisting of one word only

c) Verbs which are intransitive only
   1) consisting of more than one word
   2) consisting of one word only

d) Verbs with prepositional object or double object.

#### 2.2.3.4 New Classification

The experience gained during this year— especially through the acquisition of English translation equivalents for German entries— showed that the classification scheme set up so far was not adequate. In order to improve disambiguation, all the complement types with which a verb may occur must be listed with their semo-syntactic information. Therefore a new classification scheme was developed by the German group and is described in Section 2.3, which follows.

## 2.3 Development of a General Classification System

### 2.3.1 Purpose

As described earlier in this report, the Center's lexical lists already contain a certain amount of syntactic and semo-syntactic information. A general system of lexical features was developed which will be used to add to our established German and English noun and verb lists the information necessary for analysis and translation and in future work on the classification of German and English adjectives. Work on the establishment of the necessary feature system for adverbials will be undertaken in the coming months.

In general, two types of information are included in our feature system:

a) the properties of the classified lexical item; this information is shown as a value (or combination of values) of the subscript TY (type);

b) the properties of the environment of the lexical item; for this purpose, several subscripts and possible values are used as described below.

Note that some semo-syntactic features occur as syntactic features to facilitate encoding (cf. the subscript RL under nouns, where nouns are given the feature "may take a *when*-clause" rather than the feature "noun of time").

In general, we indicate features which represent surface phenomena. If we find, upon inspection of the completed lists, that certain features can be predicted from the occurrence of others, they will be excluded from the dictionary and introduced by means of redundancy rules.

### 2.3.2 Verb Features

Each English or German verb will be given some or all of the following subscripts. Certain of these are necessary for all verb entries; these are underlined in the list below. Others are relevant only in one of the languages we are dealing with; these are marked by G for German and E for English.

TY = type of verb (transitivity)

TS = semantic type of subject

FS = syntactic form of subject (this subscript is omitted if the verb allows only a noun phrase as subject)

$DS_G$ = deep subject (indicated only if the deep subject does not occur as a nominative in the surface sentence)

OB = syntactic form of object(s) or complement(s)

TO = semantic type of object

RA = required adverbials

OA = optional adverbials

#### 2.3.2.1 Values for Type (TY)

VT = takes at least one object which is not a reflexive pronoun

VTC = takes a cognate object only; we define a cognate object as the true cognate and all nouns subsumed under that term, as e.g., *to dance a waltz* or *a rain dance*.

VR = takes an object which must be reflexive

VT, VR = takes at least two objects, one of which must be reflexive and one which is not reflexive

VI = intransitive

NP = the verb does not passivize; verbs marked VI or VR do not need this descriptor.

$NG_E$ = the verb does not form the progressive.

2.3.2.2 Values for Type of Subject (TS)

The values which may be associated with the subscript TS are all semantic subcategories of nouns (cf. features for nouns below). In addition, the values

E = entia (any type of noun)

P = plural noun only

may be used to describe the subject a verb requires.

2.3.2.3 Values for Form of Subject (FS)

NP = noun phrase

IT = *it*

TH = *that*-clause

MI = marked infinitive

$FT_E$ = *for-to* complement

$GR_E$ = gerund

ICL = interrogative clause

$IMI_E$ = interrogative adverb + marked infinitive

$II_G$ = interrogative adverb + unmarked infinitive

2.3.2.4 Values for Deep Subject (DS)

G = genitive

D = dative

A = accusative

2.3.2.5 Values for Object or Complement Syntax (OB)

$G_G$ = genitive

$D_G$ = dative

$A_G$ = accusative

$O_E$ = noun phrase (NP) as object

all prepositions, spelled out; German prepositions which may govern the dative or accusative are marked by the numbers 1 (for accusative) or 2 (for dative), e.g., AN1, IN2, etc.

TH, MI, etc. as defined above for FS

CL = main (subjunctive) clause

PAPL = past participle

I = unmarked infinitive

BC = takes *be* + NP or ADJ

CM = takes optional *be* + NP or ADJ (e.g., *think*)

NC = takes NP complement without *be* (e.g., *elect*)

NA = takes NP or ADJ complement without *be*

AC = takes ADJ complement without *be*

2.3.2.6 Values for Type of Object (TO)

These values are all noun sub-categories (cf. noun features below), plus the values

E = entia (any type of noun)

P = plural noun only

R = reflexive

RCC = reciprocal (e.g., *aneinander geraten*)

2.3.2.7 Vaules for Required Adverbials (RA)

PLC = place (locative or directional)

DIR = direction to

ORN = origin (direction from)

TIM = time (punctual or durational)

PNC = punctual

DUR = durational

MAN = manner

MSR = measure

AC = adjective complement (for sensory verbs, as e.g., *smell good*)

2.3.2.8 Value for Optional Adverbials (OA)

The subscript OA is always associated with the same value:

DOR = direction or origin (adverb of directionality)

2.3.3 Adjective Features

Adjectives are given one or more of the following subscripts (only MD is mandatory):

TY = type of adjective

FM = form of adjective

MD = modifies nouns of the specified type

RA = requires an adverb (e.g., *wohnhaft*)

OB = form of object

TO = semantic type of object

2.3.3.1 Values for Type of Adjective (TY)

MSR = measurable (e.g., *wide* or *strong* as in *five inches wide, seven men strong*)

TM = the adjective may undergo "tough movement" (e.g., *hard, easy*)

2.3.3.2 Values for Form of Adjective

PRPL = the adjective is in form a present participle

PAPL = past participle

2.3.3.3 Values for Type of Noun Modified (MD)

All sub-categories of nouns (cf. noun features below)

TH = *that*-clause

PLU = plural, mass, or collective noun

2.3.3.4 Values for Required Adverbials (RA)

The possible values for the subscript RA are those given for the subscript RA for verbs (cf. verb features above).

2.3.3.5 Values for Form of Object (OB)

$G_G$ = genitive

$D_G$ = dative

$A_G$ = accusative

All prepositions, spelled out; case government ambiguity in German prepositions is avoided by coding 1 (accusative) or 2 (dative) after the preposition.

2.3.2.6 Values for Type of Object (TO)

The values for TO are all sub-categories of nouns (cf. noun features below), and E (any type of noun).

2.3.4 Noun Features

Nouns are semantically classified and in addition have descriptors indicating the type of attributes which they may take. The subscripts for nouns are:

TY = type of noun

SX = sex

OB = object (in case of deverbative nouns, as e.g., *dependence on*)

TO = semantic type of object

TA = takes attribute

RL = relative adverb (for deverbative nouns)

DF = derived from

FM = form (for nominalized adjectives)

2.3.4.1 Values for Type (TY)

PO = physical object

AB = abstract

AN = animate

PL = plant

IN = inanimate

HU = human

AL = animal

NM = proper name

CO = collective (components may be counted; can be used with the verb *disperse*; e.g., *group, herd, government*)

BP = body part

MS = mass (homogeneous; may occur without article in the singular; e.g., *milk, sand*)

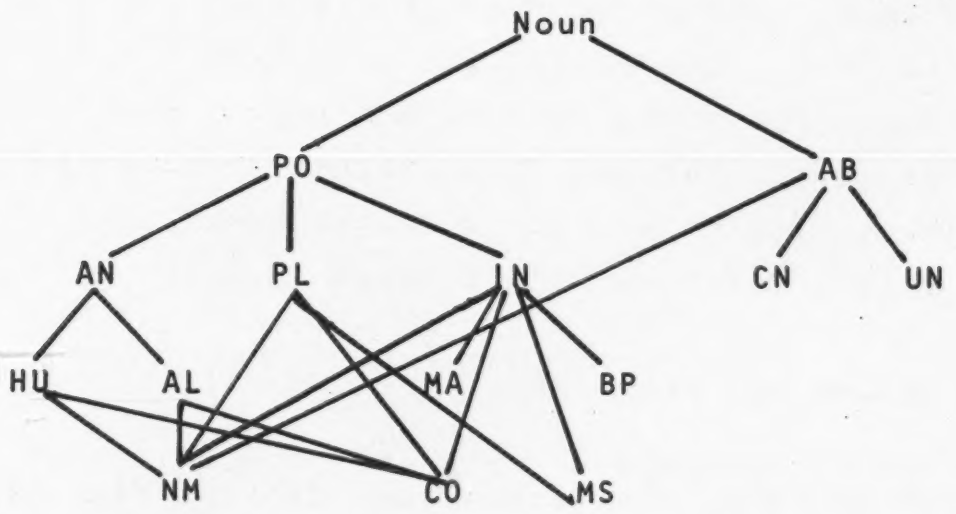
MA = machine (since they can perform some human activities)

QU = quantity (____ + (*of*) NP; e.g., *group, glass, half*, as in *a glass of milk*)

CN = count (abstract countable nouns, e.g., *idea*)

UN = unit (ADV = QUANT + ____; e.g., *mile, year*, as in *five miles long, to wait two years*)

These values may be used in combinations; e.g., the English noun *government* which has the features TY(HU CO, AB) indicating both human and collective. This value system may be represented in tree form as shown:




2.3.4.2 Values for Sex (SX)

The subscript SX has two possible values: MA (male) and FE (female).

2.3.4.3 Values for Object (OB)

The values for the subscript OB (if relevant) are all prepositions, spelled out, and followed by the numbers 1 or 2 to indicate case government when the German preposition occurs with

dative or accusative: IN1, etc..

2.3.4.4 Values for Type of Object (TO)

The possible values for the subscript TO are PO, AB, etc., as defined under TY above.

2.3.4.5 Values for Attributive (TA)

ZU = marked infinitive (e.g., *attempt*, as in *the attempt to do something*)

CL = main clause, as in *die Behauptung, dies sei die Wahrheit*

TH = *that*-clause (non-relative *that*-clauses; e.g., *his claim that this was so*)

DIR = directional adverbial complement (e.g., *a trip across Europe*)

2.3.4.6 Values for Relative Adverb (RL)

WO = where (e.g., *the place where I saw you*)

WOHIN = whereto (e.g., *the town where you went*)

WARUM = why (e.g., *the reason why he did it*)

OB = whether (e.g., *the question whether this is so*)

WIE = how (e.g., *die Frage, wie dies geschehen sei*)

ALS = when (e.g., *the time when I lived there*)

2.3.4.8 Values for From (FM)

The subscript FM may be used with only one value: A (adjective). For example, the German noun *der* (or *die*) *Abtruennige* (*the renegade*) is coded without inflectional ending and with the marker FM(A):

ABTRUENNIG TY(HU) FM (A).

# SECTION III

## *PROGRAMMING*

During this reporting period the programming effort was divided into three areas: grammar conversion programs, systems programs, and supporting programs.

### 3.1 Grammar Conversion

In order to make use of the existing IBM 7040 grammars and dictionaries it was necessary to convert them to a format suitable to the CDC 6600. The Remote File Management System (RFMS), which was being developed to facilitate management of very large data bases, was chosen. This system of programs allows the user to define a data base in tree format with no restriction on the number of branches or levels. It is based on a completely inverted file system, and the updating and retrieval features it allows are based on set theoretical operations

#### 3.1.1 Remote File Management System (RFMS F1)

The first conversion was to what will be called RFMS F1. This was simply an intermediate conversion designed to retain the information that was used by the IBM 7040 programs. The RFMS F1 Data Base definition is as follows:

```
1]LEVEL RULE NUMBER (NAME);
3]DEGREE (NAME);
4]LEFT SIDE TERM (TEXT);
6]RIGHT SIDE TERM (RG);
     62]RIGHT SIDE SYMBOL (TEXT IN 6);
     63]B OPERATOR (NAME IN 6);
     64]S OPERATOR (NAME IN 6);
7]TYPE WEIGHT INFORMATION (RG);
     71]TYPE (NAME IN 7);
     72]WEIGHT (NAME IN 7);
```

The Data Base is constructed of rules whose entries each have a component number (e.g., 3]), a name (e.g., DEGREE), and a data type (e.g., (NAME)). (RG), "repeating group", allows the following set of components to be repeated. In the above case,

each rule has only one left side but can have any number of terms on the right side.

Both the English (ENG) and German (GER) machine processable dictionaries and their syntactic and normal-form grammars were converted from IBM 7040 to RFMS F1. The ENG dictionary was made up of RMD and WEBSTER, which were in different formats.

### 3.1.2 Remote File Management System (RFMS F2)

To allow the writing of grammars containing rules in terms of complex symbols composed of subscripts, values, operators, macro statements, dummy statements, and choice statements, RFMS F2 was designed and is defined as follows:

```
1]RULE NUMBER (NAME);
2]RULE TYPES (RG);
     21]RULE TYPE (NAME IN 2);
3]DEGREE (NAME);
4]MACRO (RG);
     42]M CATEGORY SYM (NAME IN 4);
     43]M SUBSCRIPT (RG IN 4);
          431]M OP 1 (NAME IN 43);
          432]M OP 2 (NAME IN 43);
          433]M LOCATOR (NAME IN 43);
          434]M SUBSCRIPT SYM (NAME IN 43);
          435]M VALUE (RG IN 43);
               4351]M BINARY OP (NAME IN 435);
               4352]M UNARY OP (NAME IN 435);
               4353]M VALUE SYM (NAME IN 435);
          436]M SLASH (NAME IN 43);
52]L CATEGORY SYM (NAME);
53]L SUBSCRIPT (RG);
     531]L OP 1 (NAME IN 53);
     532]L OP 2 (NAME IN 53);
     533]L LOCATOR (NAME IN 53);
     534]L SUBSCRIPT SYM (NAME IN 53);
     535]L VALUE (RG IN 53);
          5351]L BINARY OP (NAME IN 535);
```

```
          5352]L UNARY OP (NAME IN 535);
          5353]L VALUE SYM (NAME IN 535);
     536]L SLASH (NAME IN 53);
54]L OP (RG);
     541]L OP SYM (NAME IN 54);
     542]L OP VALUE (NAME IN 54);
55]L CHOICE (RG);
     551]L CHOICE NUMBER (NAME IN 55);
     552]L CHOICE COMMAND (NAME IN 55);
     553]L CHOICE VALUE 1 (NAME IN 55);
     554]L CHOICE VALUE (RG IN 55);
          5541]L CHOICE VALUE 2 (NAME IN 554);
6]R SIDE (RG):
     61]R CATEGORY OP (NAME IN 6);
     62]R CATEGORY SYM (NAME IN 6);
     63]R SUBSCRIPT (RG IN 6);
          631]R OP 1 (NAME IN 63);
          632]R OP 2 (NAME IN 63);
          633]R LOCATOR (NAME IN 63);
          634]R SUBSCRIPT SYM (NAME IN 63);
          635]R VALUE (RG IN 63);
               6351]R BINARY OP (NAME IN 635);
               6352]R UNARY OP (NAME IN 635);
               6353]R VALUE SYM (NAME IN 635);
          636]R SLASH (NAME IN 63);
     64]R OP (RG IN 6);
          641]R OP SYM (NAME IN 64);
          642]R OP VALUE (NAME IN 64);
     65]R CHOICE (RG IN 6);
          651]R CHOICE NUMBER (NAME IN 65);
          652]R CHOICE OP (NAME IN 65);
          653]R CHOICE SUBSCRIPT SYM (NAME IN 65);
          654]R CHOICE VALUE (RG IN 65);
               6541]R CHOICE BINARY OP (NAME IN 654);
```

```
                6542]R CHOICE UNARY OP (NAME IN 654);
                6543]R CHOICE VALUE 2 (NAME IN 654);
7]DUMMY (RG);
     72]D CATEGORY SYM (NAME IN 7);
     73]D SUBSCRIPT (RG IN 7);
          731]D OP 1 (NAME IN 73);
          732]D OP 2 (NAME IN 73);
          733]D LOCATOR (NAME IN 73);
          734]D SUBSCRIPT SYM (NAME IN 73);
          735]D VALUE (RG IN 73);
               7351]D BINARY OP (NAME IN 735);
               7352]D UNARY OP (NAME IN 735);
               7353]D VALUE SYM (NAME IN 735);
          736]D SLASH (NAME IN 73);
     74]D OP (RG IN 7);
          741]D OP SYM (NAME IN 74);
          742]D OP VALUE (NAME IN 74);
8]TYPE WEIGHT PROBABILITY (RG);
     81]TWP ASSOCIATION NUMBER (NAME IN 8);
     82]TYPE (NAME IN 8);
     83]WEIGHT (NAME IN 8);
     84]PROBABILITY (NAME IN 8);
9]TRANSFER CROSS REFERENCE (RG);
     91]TRANSFER ROLE NUMBER (NAME IN 9);
```

The German RFMS F1 dictionary was converted to the RFMS F2 format, and work was begun toward the conflation of the incomplete English RMD and WEBSTER dictionaries and their ultimate conversion to RFMS F2.

Work was also done toward the conversion of the normal-form grammars to RFMS F2 format. As it was not possible to tell whether the interlingual substitution symbols were constructed of GER, ENG, or RUS (Russian) transfer names, it was necessary to set up a complicated conversion procedure. This involved

classifying a greater part of the 160,000 interlingual substitution symbols by hand.

When the normal-form grammars are converted, all such symbols will be reduced to their English part, and duplicate rules will be eliminated, resulting in much smaller normal-form grammars.

### 3.2 Systems Programs

The following systems programs were designed for the dictionary phase of the translation system:

a) grammar sort (DICT GS)

b) tree construction (DICT TC)

c) analysis (DICT A)

d) text display for DICT A (MATRIX)

e) choice (DICT C)

f) workspace display for DICT A and DICT C.

A subscript grammar program (SUB GRM) was designed for conversion of linguistic coding format into the full RFMS F2 format.

These programs are described below in 3.4.

### 3.3 Supporting Programs

Supporting programs were designed to:

a) update the working lexical lists (LIST UP), cf. 3.4;

b) produce new concordances (REQ CON), cf. 3.4;

c) collect statistical data;

d) automate time-consuming linguistic operations;

e) convert working lexical lists into an intermediate format for subsequent conversion into subscript format;

f) recognize poly-word entries in dictionary rules;

g) selectively display dictionary rules according to type or class name;

h) generate allomorphs for the German verb list, producing 30,000 entries from an original 17,000;

i) add class names occurring in the form prefix-stem to entries in the German dictionary;

j) convert the German noun list to an intermediate format more

amenable to updating and conversion to subscript format, i.e., each specific kind of information is assigned a specific line number.

The old grammar display program was expanded to include:

a) an analysis sort which sorts terms right-to-left, and

b) a dictionary sort with the constituents of the right-side terms concatenated.

## 3.4 Program Descriptions

### 3.4.1 Dictionary Analysis (DICT A)

Using the compiled dictionary tree constructed by DICT TC, the dictionary analysis program (DICT A) analyzes text and generates a workspace to be used by the dictionary choice (DICT C) program. The compiled tree is initially loaded onto the disk in random format and the maximum number of blocks possible is kept in memory at all times during analysis. Statistics are kept concerning the use of each block in memory. If a new block must be added, the previously loaded block with the least amount of accesses is discarded.

DICT A has two input parameters, the K-option indicator and the display indicator. If the K-option is on, the rules interpreting endings are applied everywhere except after a punctuation mark or a space. If the K-option is off, these rules are applied only after a morpheme boundary. The display indicator selects the sort option for the display of the resulting workspace. These options are from-to sorts, to-from sorts, or both.

For each file entry, the display contains the rule which applied and its number, the items "FROM" (text position where the entry begins), "TO" (text position where the entry ends), and a condition code for the application of rules interpreting the immediate right context.

The analysis program creates a table containing entries of text character sequences which match the compiled tree. Each table entry contains three items of information concerning the sequence: the location of the node in the tree, the starting character (or file), and the number of characters at this point.

The text consists of N characters (numbered from 1 to N). For each character position I, an associated file $\bar{I}$ is created which contains entries whose terminal strings end at position I. Entries for file $\bar{I}$ are referred to as FEI.1, FEI.2, etc..

Every sequence of characters defining a terminal in the tree

has as its second character a B, E, or blank represented by ° (see DICT GS). Thus at this point the node will be either a B, E, or °. If more than one character occurs at this point in the tree, these characters will be linked together by down pointers indicating branches in the tree.

For each text character processed, a new table entry is constructed if that character may begin a sequence.
Each table entry already constructed is processed as follows:

a) a new file entry is constructed if a sequence ended in the last file;

b) the table entry is updated by the new node position and the character count is either destroyed or incremented according to whether—
   (1) the sequence does or does not continue as part of another rule, and
   (2) the second character of the sequence is or is not being processed;

c) the starting branch conditions are evaluated (as opposed to character matches being performed as in the cases above), if—
   (1) a sequence continues as part of another rule, and
   (2) the second character of the sequence is being processed.

If the second character of the sequence being processed is:
B, the string may not begin if the previous file—
   (1) does not contain an interpreted string,
   (2) contains a punctuation mark, or
   (3) contains a blank;

E, the string may begin;

°, the string may not begin if the previous file does not contain an interpreted string.

During the processing of the second character, the first reference to the table entry modifies the entry. All future references create new table entries. After all table entries for the character are processed, the table is resorted to put the longest sequence first, if and only if there were any multiple second-character table-entry constructions.

As each new file entry is constructed, the left-side operators M, ¬, and ° are used to compute the value for the FROM file. This value will be used by the following file to determine whether the new file may be constructed. If the second character is a P and the value of the previous file indicates a blank or punctuation mark, a new file entry is completed.

### 3.4.2 Dictionary Choice (DICT C)

DICT C processes the from-to workspace output from DICT A. It discards all file entries from the workspace which do not belong to a sequence of rules which span M-symbols. (The M-symbols are, primarily, blanks or punctuation marks, including hyphens.) It also generates K-rules for all M-symbol sequences which are not spanned. It has four input parameters. The first sets the K-option either on or off (cf. 3.4.1). The second sets the preference-(P-)option either on or off. The third records which workspace display is requested for output— the options being any choice or combination of: to-from, from-to, or, all deleted file entries. The fourth parameter indicates whether the from-to workspace should be saved or destroyed. (Word analysis uses workspace in the to-from format.)

DICT C reads in file entries until it finds a group which completely spans two M-symbols. It processes this group and then reads in the next group.

The first operation performed is the elimination of all file entries from this group which have right-side F-operators and are not followed by an M-symbol. An F-operator is assigned to all rules for which only punctuation or ° can follow.

If the P-option is on, all other sequences or file entries covering the same span are discarded from any file entry having a left-side P-operator. The P-operator in a rule gives preference to a long span over two or more short spans.

The rules used in all possible sequences covering the span are tagged for later processing. Processing for this span is terminated when a possible sequence is found without M-symbols resulting from a rule with a multi-word right-side. If a possible sequence with an internal M-symbol is found, all possible sequences are calculated for each subspan. If a subspan is not completely covered, a K-rule is generated. When the K-option is on, additional K-rules are constructed which link together all possibilities for prefixes and suffixes.

If the original span was not covered, a K-rule is generated to cover it. Additional K-rules are also generated for sequences of the form: prefix-K, prefix-K-suffix, and K-suffix; and each of these sequences covers the original span.

### 3.4.3 Dictionary Grammar Sort (DICT GS)

DICT GS has two major functions— a) to determine the restrictions on the application of a particular rule, and b) to

sort the dictionary grammar according to the right-side term(s) and the application restriction information (ARI).

The dictionary contains sixty-four roots. From each root four branches may theoretically extend which represent the restrictions for all terminals. These branches are the [P]-restriction, the [°]-restriction, [B]-restriction, and [E]-restriction. The [P]-restriction indicates that the rule may apply to a string which is preceded by a punctuation mark or blank. Both the [°]-restriction and the [B]-restriction indicate that the rule may apply to a string which is contiguous to a preceding interpreted string. The [B]-restriction also indicates that the span must not be preceded by a punctuation mark or blank. The [E]-restriction indicates that the rule may apply anywhere; there are no restrictions in this case.

To construct a grammar tree, the ARI of the rules needs to be retained. Therefore, depending upon the ARI in the rule, the program DICT GS inserts a "B", "E", or "°". (The [P]-restriction is included under the °-indicator at this point. In the surface dictionary analysis, a distinction is made.)

DICT GS strips RFMS loader-format repeating-group names and extraneous information from the rule. The program generates sort keys, consisting of the right-side terms and the ARI, and retains the left-side terms and ARI as data. The ARI indicator is the second character in the sort key.

Each rule in the dictionary grammar is converted to the following form in DICT GS—

| | |
|---|---|
| Word 1: | Length of rule (revised for SORT/MERGE routine), M, and length of sort key, N; |
| Words 2 → (N+1): | Sort key; |
| Words (N+2) → M: | Sort data area. |

The program then sorts the rules in the dictionary grammar, which are in the form listed above.

Finally, DICT GS creates a new tape consisting of two records. The first record contains information concerning the length of the longest sort key created, the length of the longest data area created, and the date the new tape was created. The second record contains the sorted dictionary grammar. This new file is used as input to the dictionary tree construction program.

### 3.4.4 Dictionary Tree Construction (DICT TC)

DICT TC builds the compiled dictionary tree and its index

from the output of DICT GS. It reads in one entry at a time, comparing it character by character with the previous entry. Where the character strings differ, a down pointer is attached to the previous string to indicate the place where the new string continues. If the old string is a subset of the new string, a continuation (or right) pointer is attached to the end of the old string. In both cases, after all the characters are placed in the tree, the remaining information (e.g., the rule number and the left-side of the rule) is added at the end of the string. Another new entry is read in and the process is repeated. Each time a new first character is encountered, a pointer is placed in the index table. Thus, after the process is completed, there is a pointer to the beginning of every character tree. The index and the compiled tree are then written out in a form suitable for use by DICT A.

### 3.4.5 Subscript Grammar (SUB GRM)

SUB GRM converts subscript rules from the form in which the linguists encode them into RFMS F2 Loader Input format. Rule numbers and duplication numbers are optional input. All rules containing format errors are discarded.

### 3.4.6 List Update (LIST UP)

LIST UP updates all the working lexical lists. These are in the form of card images, each of which is indexed by corpus, request, and line numbers. LIST UP allows additions, deletions, insertions, and replacements on a card-for-card basis. The output consists of all requests plus all changes, or only those requests for which a change was made, and a new updated tape.

### 3.4.7 Concordance Program (REQ CON)

A new concordance program was constructed having the following features:

a) A display of the concorded word in the context of the entire request (identified by the digits in columns 4-7)

b) Forward and/or backward sorts. Each sort includes all the words in the request. In the forward sort the request, in the following succession, is used to determine the list order for the concorded word—

1) concorded word

2) words in sequence to the right of the concorded word

3) a "zero word", inserted at the end of the request, which takes precedence over any other word at the same point

4) words in sequence to the left of the concorded word.

A backward sort takes the words to the left first, and then the words to the right, inserting the "zero word" at the beginning of the request.

c) An inclusion/exclusion option

d) A glossary of all concorded words with their frequencies

e) A choice of no display or any of three forms of output display— all the requests, only those requests which were used, or, the requests not used

f) Standard or non-standard procedure for concording words. Standard is based on the occurrence of the word itself; non-standard refers to words preceded by a special character. For the latter, pre-processing programs for tagging the words to be concorded may be required.

g) Concordance restrictable to specified sequences of starting characters. This capability permits the recovery of information when the capacity of the computer is exceeded.

## *CONCLUSION*

Progress under the contract has been good, in spite of reduced funding. The theory underlying the Linguistics Research System has been developed. The linguistic descriptions which are necessary to implement this theory, however, have not met our original projections, because of lack of manpower. Programming has also suffered from the reduction in funding.

During the remainder of the contract period a lexicon will be produced which will have "precise information on the syntactic and semantic properties of lexical items". Preliminary grammars as required for the implementation of the Linguistics Research System will be produced.

Much of the programming effort has been concerned with bringing our linguistic data into the formats required by the Linguistics Research System, and with updating the Center's lexical data bases. In the last two years of work under the contract, programs will be constructed for handling the grammars described in Section I of this report and the German and English lexical data.

## *REFERENCES*


1. Bobrow, D.G., and J.B. Fraser. 1968. "An augmented state transition network analysis procedure," in *Proceedings of the International Joint Conference on Artificial Intelligence*. Washington, D.C..

2. Chomsky, Noam. 1965. *Aspects of the Theory of Syntax*. Cambridge, Mass.: M.I.T. Press.

3. Earley, Jay. 1970. "An efficient context-free parsing algorithm." *Communications of the ACM* 13:94-102.

4. Hornby, A.S., E.V. Gatenby, and H. Wakefield. 1963. *The Advanced Learner's Dictionary of Current English*. 2nd ed. London: Oxford University Press.

5. Lehmann, W.P., and Rolf A. Stachowitz. 1970. *Research in German-English Machine Translation on Syntactic Level*. Vol. II. (RADC-TR-69-368.) Austin: Linguistics Research Center, The University of Texas at Austin.

6. _____. 1972. *Normalization of Natural Language for Information Retrieval*. Final Technical Report. Austin: Linguistics Research Center, The University of Texas at Austin.

7. Petrick, Stanley R. 1965. *A Recognition Procedure for Transformational Grammars*. Ph.D. dissertation, M.I.T..

8. _____. 1971. "Syntactic analysis for transformational grammars," in *Feasibility Study on Fully Automatic High Quality Translation*, by W.P. Lehmann and Rolf A. Stachowitz. Austin: Linguistics Research Center, The University of Texas at Austin.

9. Thorne, J., P. Bratley, and H. Dewar. 1968. "The syntactic analysis of English by machine," in *Machine Intelligence 3*, D. Michie, ed.. New York: American Elsevier.

10. Walker, D.E., P.G. Chapin, M.L. Geis, and L.N. Gross. 1966. "Recent Developments in the MITRE Syntactic Analysis Procedure." Bedford, Mass.: The MITRE Corp.

11. Walker, D.E., P.G. Chapin, M.L. Geis, and L.N. Gross. 1966. "Recent Developments in the MITRE Syntactic Analysis Procedure." Bedford, Mass.: The MITRE Corp..

12. *Webster's New Collegiate Dictionary*. 1960. Springfield, Mass.: G. & C. Merriam Co..

13. Wildhagen, Karl, and Will Héraucourt. 1953, 1959. *English-German German English Dictionary*, 2 vols. London: George Allen & Unwin; Wiesbaden: Brandstetter.

14. Woods, W.A. 1970. "Transition network grammars for natural language analysis." *Communications of the ACM* 13: 591-606.

15. Zwicky, A.M., D.E. Walker, J. Friedman, and B.C. Hall. 1965. "The MITRE syntactic analysis procedure for transformational grammars," in *Proceedings of the 1965 Fall Joint Computer Conference*. New York and Washington, D.C.: Spartan.

UNCLASSIFIED
Security Classification

## DOCUMENT CONTROL DATA - R & D

*(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)*

1. ORIGINATING ACTIVITY *(Corporate author)*
University of Texas at Austin
Linguistics Research Center
Austin, Texas 78712

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b. GROUP
N/A

3. REPORT TITLE
DEVELOPMENT OF GERMAN-ENGLISH MACHINE TRANSLATION SYSTEM

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
Technical Report (First Annual) 1 February 1970 - 31 January 1971

5. AUTHOR(S) *(First name, middle initial, last name)*
Dr. Winfred P. Lehmann
Dr. Rolf A. Stachowitz

6. REPORT DATE
March 1972

7a. TOTAL NO. OF PAGES
50

7b. NO. OF REFS
15

8a. CONTRACT OR GRANT NO.
F30602-70-C-0118

Job Order No. 45940000

9a. ORIGINATOR'S REPORT NUMBER(S)
None

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
RADC-TR-72-47

10. DISTRIBUTION STATEMENT
Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES
None

12. SPONSORING MILITARY ACTIVITY
Rome Air Development Center (IRDT)
Griffiss Air Force Base, New York 13440

13. ABSTRACT

The report presents progress in theoretical linguistics, descriptive linguistics, lexicography, and systems design in the development of a German-English Machine Translation System. Work in the theoretical group concentrated on intrasentential disambiguation and on improving certain parts of the system to achieve greater economy in processing. The linguistic group was engaged in correcting and updating the existing German and English lexical data bases by assigning syntactic and semantic selection restrictions to lexical items. Work in the system group concentrated on the reduction of the size of the existing LRS lexical data base without information loss, on the conversion of this data base to the LRS subscript format, on the construction of supporting programs to expedite and facilitate the updating of the LRS word lists, and on the construction of part of the LRS grammar maintenance and systems programs.

DD FORM 1 NOV 65 1473

UNCLASSIFIED
Security Classification

UNCLASSIFIED
Security Classification

| 14. KEY WORDS | LINK A | | LINK B | | LINK C | |
|---|---|---|---|---|---|---|
| | ROLE | WT | ROLE | WT | ROLE | WT |
| Linguistic Theory | | | | | | |
| Computational Linguistics | | | | | | |
| Machine Translation R&D | | | | | | |
| Research in Syntax/Semantics | | | | | | |
| Lexicography in Machine Translation | | | | | | |
| Systems Design in Machine Translation | | | | | | |

SAC--Griffiss AFB NY

UNCLASSIFIED
Security Classification